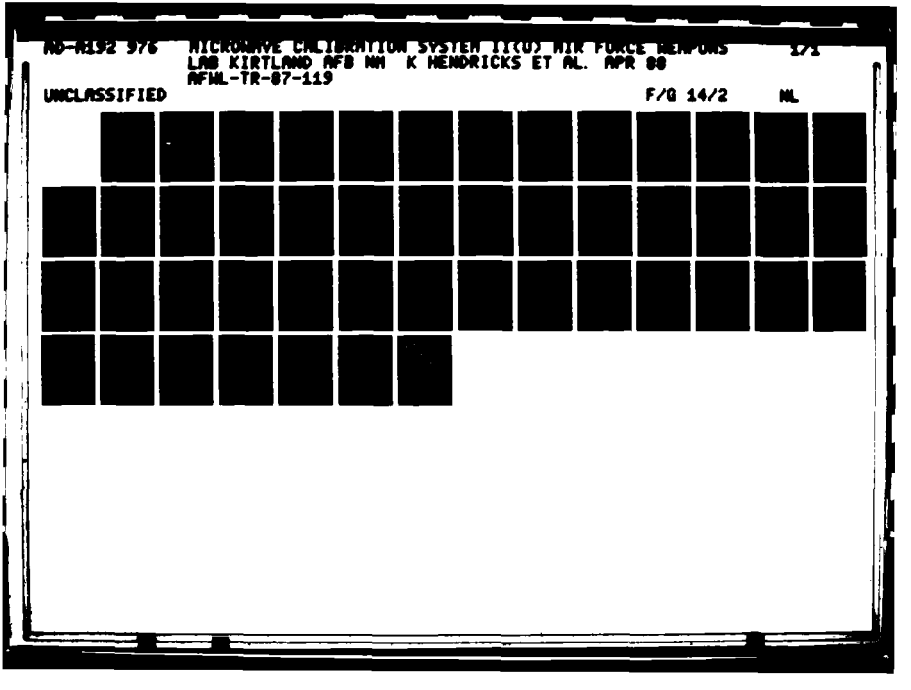
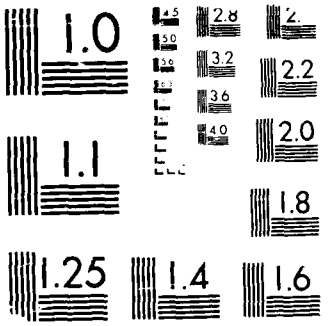

AD-A192 976 MICROWAVE CALIBRATION SYSTEM II(U) AIR FORCE WEAPONS 1/1
LAB KIRTLAND AFB NM K HENDRICKS ET AL. APR 88
AFWL-TR-87-119
UNCLASSIFIED F/G 14/2 NL



MICROCOPY RESOLUTION TEST CHART

AFWL-TR-87-119

AFWL-TR-
87-119

(2)

MICROWAVE CALIBRATION SYSTEM II

K. Hendricks
G. DeMuth
L. Torraca

April 1988

Final Report

AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base, NM 87117-6008

38

This final report was prepared by the Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico under Job Order 57972401 . Captain Kyle J. Hendricks (AWPB) was the Laboratory Project Officer-in-Charge.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been authored by employees of the United States Government. Accordingly, the United States Government retains a nonexclusive, royalty-free license to publish or reproduce the material contained herein, or allow others to do so, for the United States Government purposes.

The Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

If your address has changed, if you wish to be removed from our mailing list, or if your organization no longer employs the addressee, please notify AFWL/AWPB, Kirtland AFB, NM 87117-6008 to help us maintain a current mailing list.

This report has been reviewed and is approved for publication.

Kyle J. Hendricks
KYLE J. HENDRICKS
Capt, USAF
Project Officer

FOR THE COMMANDER

Bruce R. Anderson
BRUCE R. ANDERSON
Maj, USAF
Chief, Beam Physics Branch

William L. Baker
WILLIAM L. BAKER, PhD
Chief, Advanced Technology Division

DO NOT RETURN COPIES OF THIS REPORT UNLESS CONTRACTUAL OBLIGATION OR NOTICE ON A SPECIFIC DOCUMENT REQUIRES THAT IT BE RETURNED.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFWL-TR-87-119	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Air Force Weapons Laboratory 6b OFFICE SYMBOL (If applicable) AWPB	7a NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) Kirtland Air Force Base, NM 87117-6008	7b ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO
62601F	5797	24	01

11 TITLE (Include Security Classification)
MICROWAVE CALIBRATION SYSTEM II

12 PERSONAL AUTHOR(S)
Hendricks, K., DeMuth, G., and Torraca, L.

13a TYPE OF REPORT	13b TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Final	FROM 3/87 TO 8/87	1988, April	50

16. SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Microwave, Rf, Radiofrequency, Calibration, Frequency Response, Power Response Detector, Component
20	14		
14	02		

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

An expanded microwave calibration system is presented. This report discusses the process used to obtain a calibration for a detector, and the various hardware and software pieces for the system. Options available in using the system are presented.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	Unclassified

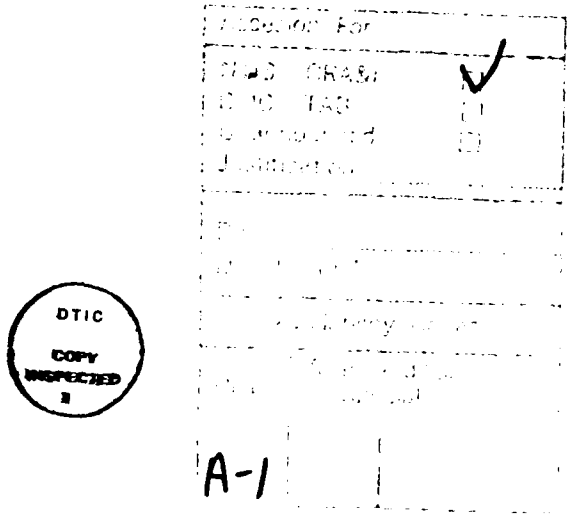
22a NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE (Include Area Code)	22c OFFICE SYMBOL
Kyle J. Hendricks	505 844-0121	AWPB

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

CONTENTS



FIGURES

INTRODUCTION

This report documents changes and additions to a previous microwave calibration system report (Ref. 1). In brief, the changes made to the old system include the system controller and software drivers for the devices on the GPIB (IEEE-488) bus, and the additions are the data reduction software used to generate tables and graphs from the calibration data.

This report first discusses the reason for computerizing the calibration process. Second, the hardware components necessary to complete the calibrations are listed. Finally the programs used to obtain and reduce the raw data are discussed.

PURPOSE

The primary reason for using a computerized calibration system was to reduce the amount of time required to obtain the calibration and get useful information from it. Initially, the frequency response was obtained from an oscilloscope and the data read from a Polaroid photo. The process of obtaining the photo was quick, but reading the data was time consuming and many errors could be introduced. When the time for each calibration was then multiplied by the number of calibrations to be completed, it was apparent that too much time would be spent doing calibrations.

All that is needed from the calibrations are frequency responses from various components, the only exception being crystal diode detectors where applied power variation is also needed. Obtaining these variations is simple for a program that loops on the frequency and/or power and measures the response of the component under test. This is the basis for all the calibration acquisition software.

The problem with the above acquisition scheme is the frequency response of the various monitor components (i.e., power sensors and directional couplers)

---

1. Hendricks, K., Microwave Calibration System, AFWL-NTYP-TN-84-004, Air Force Weapons Laboratory, Kirtland Air Force Base, N Mex, July 1984.

to microwaves. The solution to this problem is to have the reduction software read the raw data files and remove the response of the power sensors and any directional couplers used to obtain the raw data. This process gives a better method of knowing what signal had to be applied to a component in order to get the measured data.

A side benefit from completing our calibrations with this hardware is portability. The calibration cart can be used anywhere in the laboratory area and calibrate devices in place. This provides better calibrations and also allows for modifications of experiments without having to completely disassemble and rebuild the experiment with independently calibrated hardware.

HARDWARE

The hardware used to obtain the calibration data works under the premise that, if enough data points are used, one can adequately approximate the real response function. The hardware (Fig. 1) necessary to accomplish this tabulation of data includes: (1) a controller (Z100 microcomputer) capable of large amounts of data storage; (2) an oscillator (HP 8350A) with variable power and frequency; (3) various power sensors (HP 8478B, 8481A, 8484A) to measure the applied power to the device under test; (4) an interface [a BIRA analog-to-digital converter (ADC) housed in a CAMAC crate] to transfer the data from the power sensors to the storage device; and (5) a computer (VAX 11/750) to take all the various sized files and unfold the imbedded sensor response funtions.

The basic change to the above list is the controller. Previously a Tektronix 4051 microcomputer, with GPIB and RS-232C interfaces, was used. We are now using a Zenith Z-100 microcomputer with a built-in RS-232C interface and an add-on National Instruments GPIB interface. The software driver was purchased from National Instruments for ZBASIC running under the MS-DOS operating system.

The reason for using the Z100 instead of the 4051 was its increased storage ability. The Z100 is equipped with a 10-megabyte Winchester drive and a

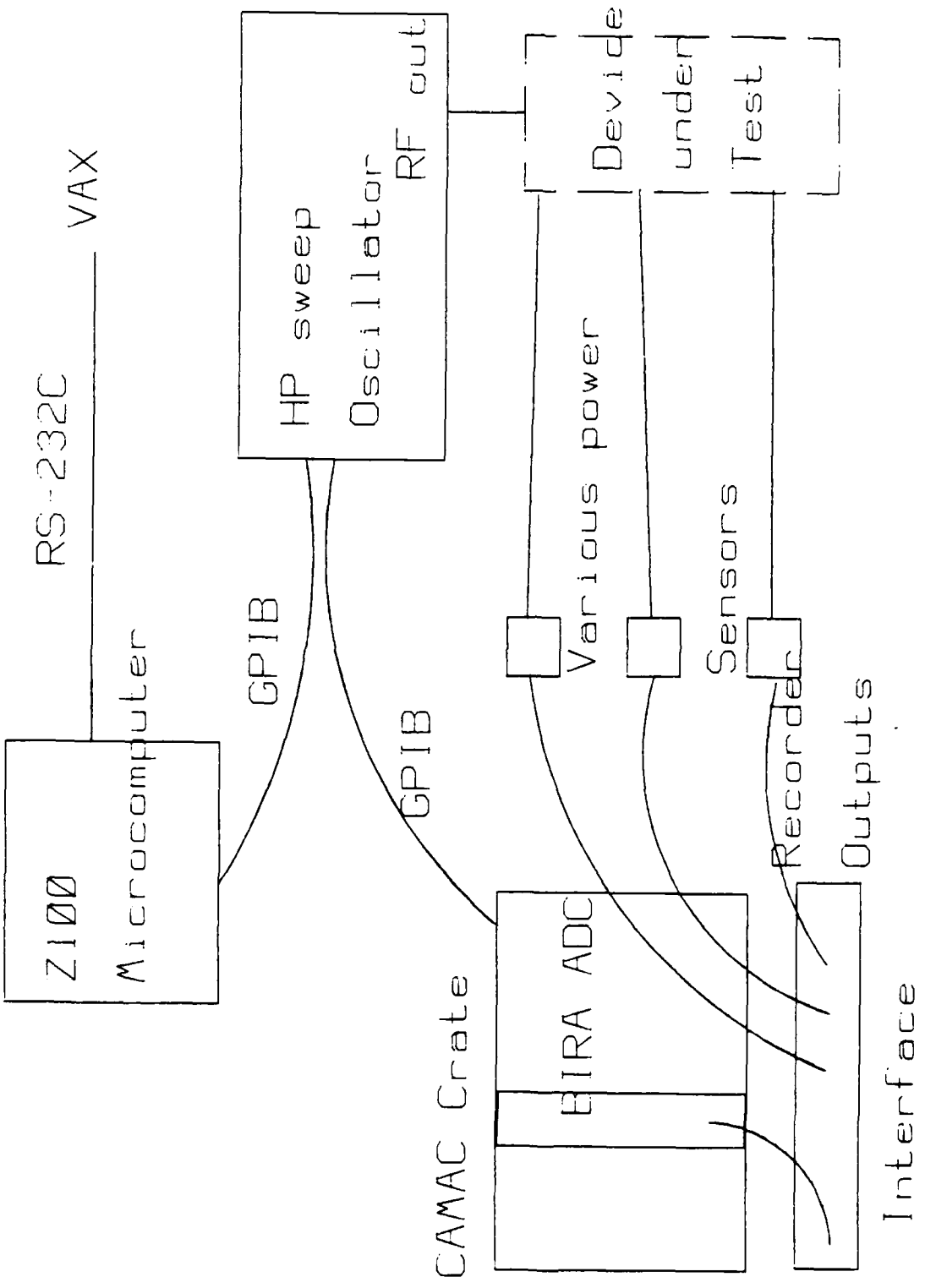


Figure 1. Z100 calibration system.

360-kilobyte floppy drive. The programs are maintained on the Winchester disk, and the data files are kept on various floppy diskettes for crystal, attenuators, cables, and other designated diagnostics.

The one shortfall of the Z100 is the limitation in memory space for data storage. The amount of data required by the reduction software typically exceeds the 64-kilobyte limit and as a result, our reduction software is run on the VAX 11/750. Having to use the VAX for data reduction requires extra time when the calibration cart must move about the lab. However, since many IBM-AT type computers are available (with terminal software), one can get to the VAX by taking the floppy disk to any of the microcomputers.

SOFTWARE

Z100

The software for the Z100 is essentially the same as that used on the 4051, modulo language differences. We have moved the software for the crystal calibrations, directional coupler calibrations, and "attenuation" calibrations. Hookup diagrams are shown in Figs. 2, 3, and 4.

Figure 2 indicates the intended setup for the program DCCAL.BAS. The through-power sensor's recorder output is hooked to port 1 of the BIRA ADC bus. The pickoff-power sensor's recorder output goes to port 2 of the BIRA ADC bus.

Figure 3 indicates the hookups for measurements made using the program DB.BAS. The incident-power sensor is attached to port 1 of the BIRA ADC, the reflected-power sensor is attached to port 2 of the BIRA ADC, and the transmitted-power sensor is attached to port 3 of the BIRA ADC.

Figure 4 indicates the proper set up for using XTAL.BAS. The incident and reflected sensors are connected as for DB.BAS. The crystal must be calibrated by driving the same load it sees during actual experiments. In this work, that is 50 Ω. An artifact of driving a 50-Ω load is to cause the voltage level to be between 0 and -100 mV. This voltage is to be measured by

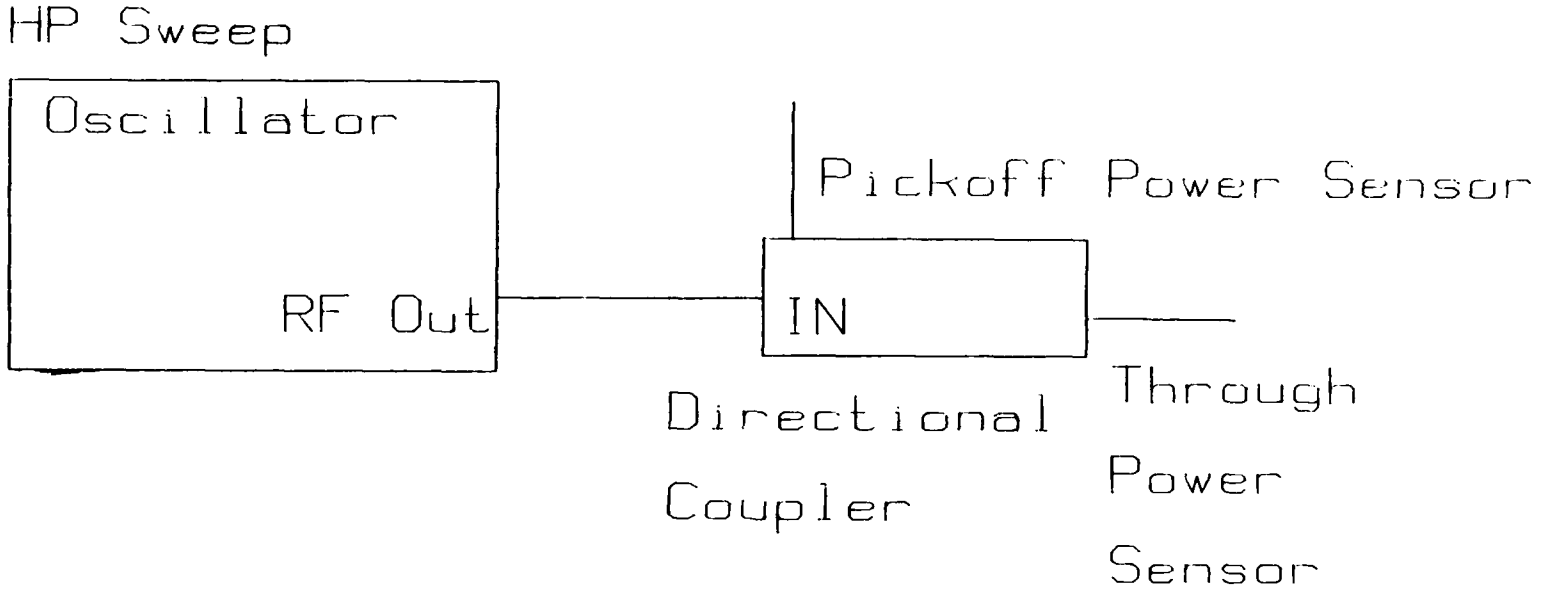


Figure 2. Directional coupler calibration.

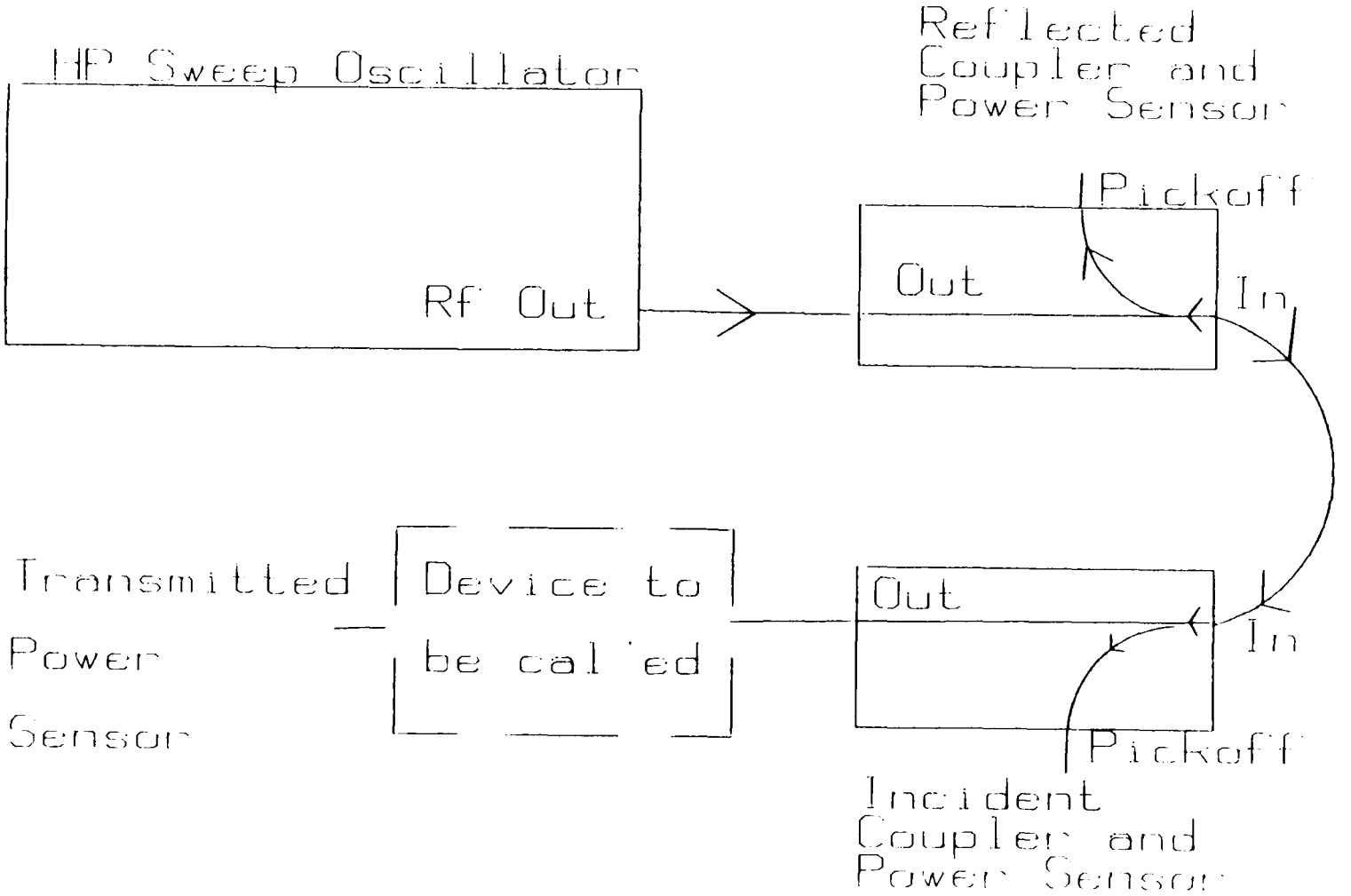


Figure 3. Attenuation calibration.

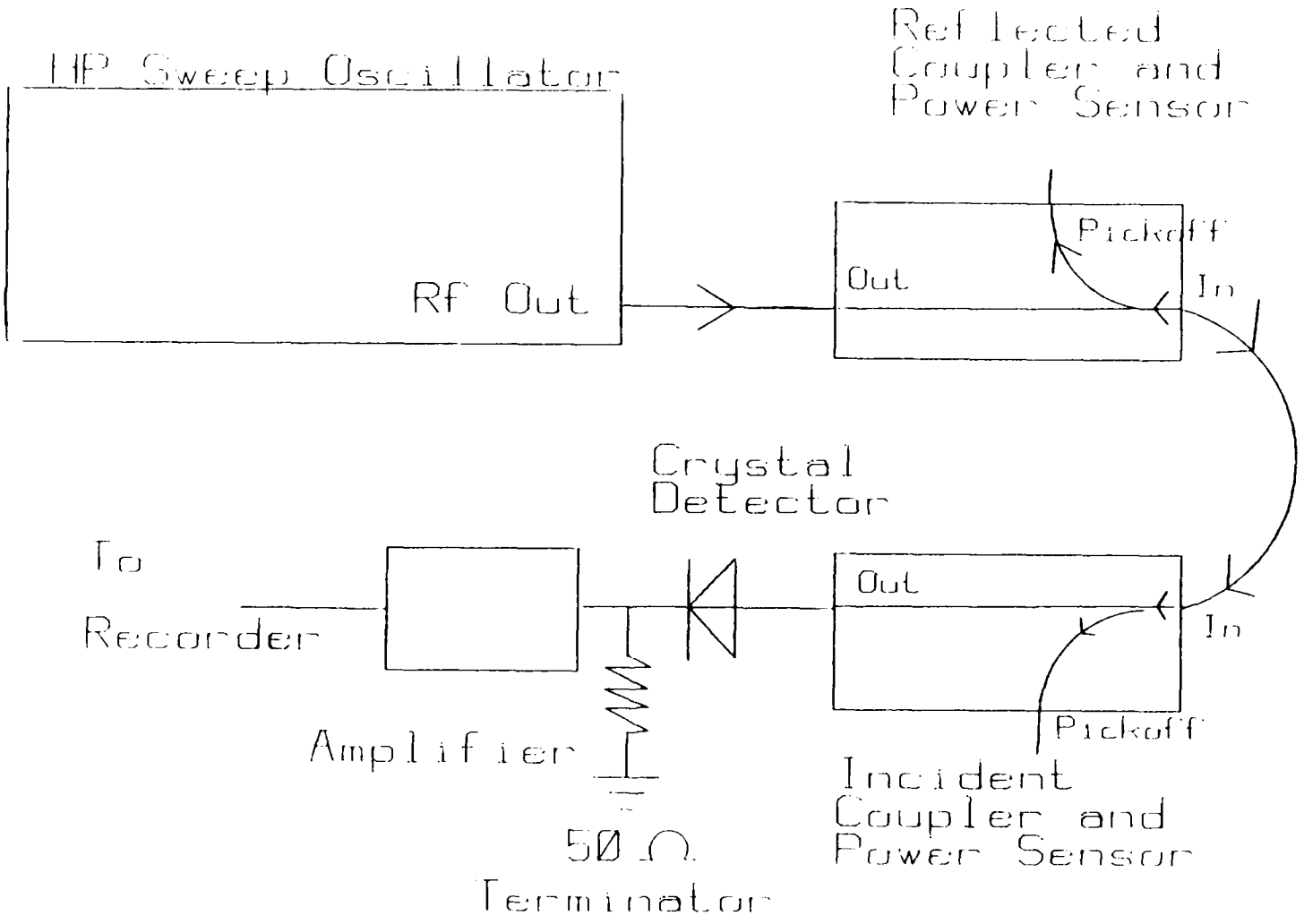


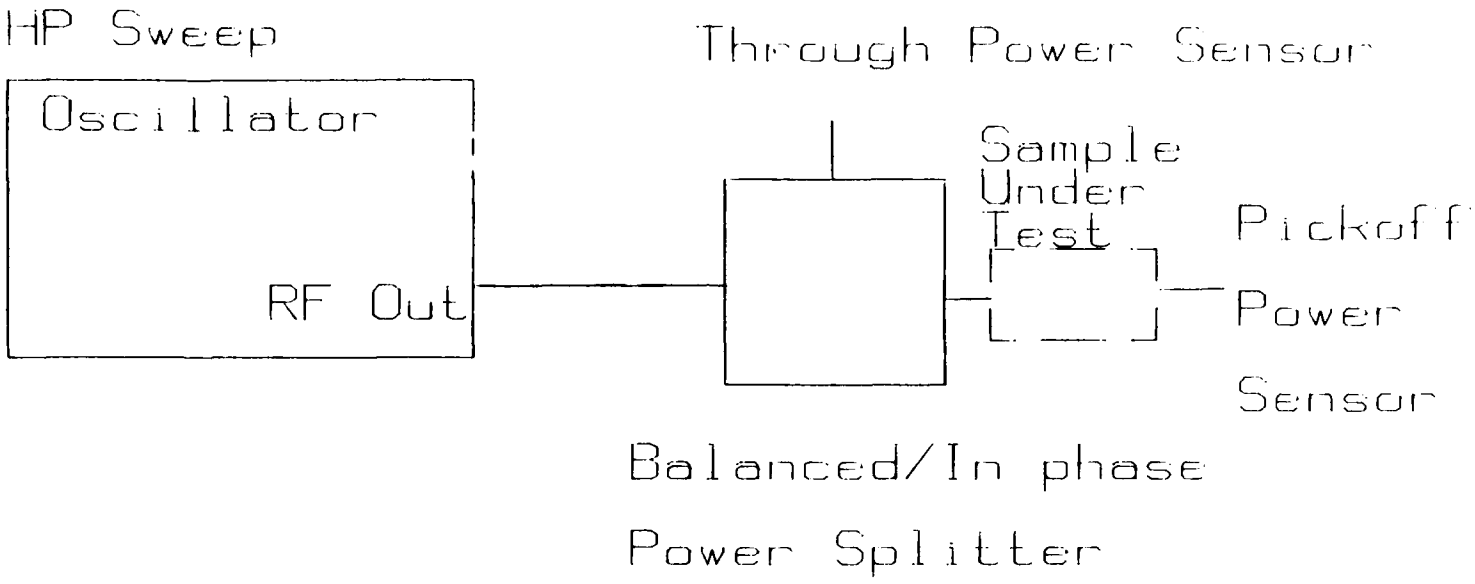
Figure 4. Crystal diode detector calibration.

an ADC with 12 bits of resolution for a 0- to +5-V range. To get around using only the low bits of the ADC and the polarity problem, a simple inverting amplifier circuit (using a 741 operational amplifier) is used. The gain is set to ~ 58 (actually 35.3 dB ± 0.2 dB = 20. * log $|V_{out}/V_{in}|$) over the output limits of the sweep oscillator. The gain is then divided out in the program and the quotient is stored as the measured value.

These programs have considerable flexibility built into them. An example is shown in Fig. 5. This setup was used to calibrate cables outside the frequency range of the directional couplers. Balanced power splitters are available which may be used to DC frequencies. The splitter was first calibrated and found to have 0 dB (± 0.2 dB) relative attenuation between outputs; this meant the splitter could obtain directly the frequency response of cables, or any other component response down to 10 MHz (the limit of the sweep oscillator).

Program listings are included in the appendix. As noted in Ref. 1, segments in the programs can change the parameter range of the calibration. The parameters one may want to change are: (1) power level or range of power levels applied, (2) frequency range covered, (3) number of data points measured both in frequency and power, and (4) power meter scale factors. In the program DCCAL.BAS, line 1130 specifies the start frequency (F1) and stop frequency (F2), the applied power level (P0), and the number of data points (N0). Also, in line 1227 the power meter scale factors are preset. Parameter M(1) is the scale factor needed to convert the voltage output read on ADC port 1 to the indicated power level on the meter, and M(2) is the scale factor for ADC port 2.

The scale factors are determined in the following manner. The power meter manuals state that the recorder output is a DC voltage between 0 and 1.2 V regardless of power meter scale. For example, a voltage of 0.9 V is output for 90 μW on the 100-μW scale, 900 μW on the 1-mW scale, and 9 mW on the 10-mW scale. The scale factor is that number which, when multiplied with the voltage from the recorder output, gives a power level in mW. In the case of 90 μW, the scale factor would be 0.1; for 900 μW the scale factor would be 1; and for 9 mW the scale factor would be 10.



This calibration uses DCCAL.BAS

Figure 5. Alternative attenuation calibration.

The parameter block in ATTN.BAS is found at lines 1375 to 1430. The parameter CO is a counter used for delay loops and can be ignored. The parameter block for XTAL.BAS is found at lines 1490 to 1580. The additional parameters are P1 and P2, the start and stop power level, N1 the number of power levels to take data, YMAX a typical maximum voltage output by the crystal diodes, and G1 the gain of the amplifier box. All other parameters are requested by the program and need no explanation.

VAX

The programs for the VAX are also included in the appendix. Notice that for the files accessed by programs DB.FOR and XTAL.FOR, one can have applied power computed in two ways: (1) using only the measured incident power; and (2) using the difference of the incident and reflected power. Case (1) is more useful for pulsed microwave measurements. Case (2) is more useful in CW microwave measurements.

These programs use the subroutines DCCOR and TCOR to remove the frequency response of the power sensors and calibrated directional couplers used in the data acquisition to calculate the applied, reflected, or transmitted power signal. These subroutines work from data files and must interpolate between the data points. The data is interpolated by a cubic spline method to give better accuracy than a simple linear interpolation.

The program DRC.FOR uses the output file from DCCAL.BAS to give frequency response of directional couplers, or cables. The output of DRC.FOR is a file DRATTN.Pat which should be renamed to indicate the calibrated device.

The program DB.FOR uses data files created by ATTN.BAS. The reduced data are saved in a file called SAMPLE.DAT. This file should also be renamed for later use.

The programs XTAL.FOR and PSWP.FOR are used to get calibration data from XTAL.BAS. XTAL.FOR reads the raw data file from XTAL.BAS and folds out the frequency response of the directional couplers and power sensors to get back to the actual power applied to the crystal. After one decides how the

applied power is to be calculated, the files XTALV.DAT and XTALP.DAT are created. These files contain information on the voltage and power measured in matrix form. The columns of each file are "constant" power applied, and the rows are "constant" frequency applied. These files should be renames for later use. The renamed files are used by PSWP.FOR to give a power sweep at a chosen frequency. If the chosen frequency is not one of the measured frequencies, the data in both files are interpolated by computing a cubic spline on the frequencies immediately above and below and a computed set of numbers are generated to fill in XTALV and XTALP. These computed numbers are then used to give back to the user the power level necessary to generate the measured voltage.

In running all the data reduction software, graphs are generated at many places that are useful for later quick calculations or when looking for "bad" data. A hardcopy might be helpful. The programs also create data files which are tables of the data used to draw the graphs. These files are useful when more accurate information is needed for calculations.

CONCLUSIONS

This report has discussed an improved microwave calibration system. The three areas covered were: (1) the hardware used to build the calibration system; (2) the software used to acquire the raw calibration data; and (3) the software used to reduce the raw data and obtain the "true" calibration response functions of our microwave detectors.

The system does operate as advertised on the date of this report. Modifications to the system may occur in the future, and will be properly documented at that time.

APPENDIX
PROGRAM LISTINGS

DCCAL.BAS

```
10        CLEAR    ,60000!          ' BASIC Declarations
20        IBINIT1 = 60000!
30        IBINIT2 = IBINIT1 + 3
40        BLOAD "bib.m",IBINIT1
50        CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,
                       IBLOC,IBPPC,IBBNA,IBONL,IBRSC,
                       IBSRE,IBRSV,IBPAD,IBSAD,IBIST,
                       IBDMA,IBEOS,IBTMO,IBEOT,IBRDF,IBWRTF)
60        CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,
                       IBWRTA,IBCMD,IBCMDA,IBRD,IBRDA,
                       IBSTOP,IBRPP,IBRSP,IBDIAG,IBXTRC,
                       IBRDI,IBWRTI,IBRDIA,IBWRTIA,IBSTA%,
                       IBERR%,IBCNT%)
1000 CLS
1010 WRT$="SWEEPER":CALL IBFIND(WRT$,SWEEPER%)
1020 WRT$="CAMAC":CALL IBFIND(WRT$,CAMAC%)
1030 WRT$="GPIB0":CALL IBFIND(WRT$,GPIB0%)
1040 CALL IBCLR(SWEEPER%)
1045 REM TURN RF OFF AND
1046 REM TURN CW FILTER AND INTERNAL LEVELING ON
1050 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
1052 WRT$="A1":CALL IBWRT(SWEEPER%,WRT$)
1054 WRT$="FL1":CALL IBWRT(SWEEPER%,WRT$)
1060 REM HEADER BLOCK ENTRY
1070 PRINT "ENTER THE FILENAME"
1080 INPUT V$
1090 OPEN "O",#1,V$
1100 PRINT "ENTER TEST LABEL (IN SINGLE QUOTES)":INPUT V$
1110 I0=0:WRITE #1,I0
1120 DIM D(3),M(2)
1125 REM DEFINE FREQUENCY RANGE, POWER LEVEL,
1126 REM AND NUMBER OF DATA POINTS
1130 F1=.5:F2=12!:P0=11!:N0=30
1140 PRINT #1,V$;,""POWER LEVEL (DBM)"";P0
1145 REM SET FREQUENCY STEP SIZE
1150 LET S0=(F2-F1)/(N0-1)
1160 PRINT "ENTER ID FOR DIRECTIONAL COUPLER":INPUT D(3)
1170 PRINT "ENTER ID FOR THROUGH THERM MOUNT":INPUT T1
1180 PRINT "ENTER ID FOR PICKOFF THERM MOUNT":INPUT T2
1190 REM SEND HEADER TO 11/60
1200 PRINT #1,""DIRECTIONAL COUPLER ID#"":WRITE #1,D(3)
1210 PRINT #1,""THROUGH THERMISTOR MOUNT ID#"":WRITE #1,T1
1220 PRINT #1,""PICKOFF THERMISTOR MOUNT ID#"":WRITE #1,T2
1225 REM PRESET POWER METER SCALE FACTORS
1227 M(1)=10:M(2)=1!
1230 PRINT "THE FOLLOWING SETUP IS USED"
1240 PRINT "THE THROUGH POWER IS MONITORED ON PORT 1"
1250 PRINT "THE PICKOFF POWER METER IS ON A SCALE OF ";M(1)
1260 PRINT "THE PICKOFF POWER METER IS ON A SCALE OF ";M(2)
1270 PRINT "ENTER A 'Y' WHEN SET"
```

```
1280 INPUT V$
1290 IF V$=CHR$(89) OR V$=CHR$(121) THEN 1310 ELSE 1280
1310 PRINT #1,''FREQ(HZ) THROUGH  POWER(W)  PICKOFF POWER(W)''
1320 CALL IBCLR(SWEEPER%)
1330 WRT$="FA":CALL IBWRT(SWEEPER%,WRT$)
1340 WRT$=STR$(F1):CALL IBWRT(SWEEPER%,WRT$)
1350 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1360 WRT$="FB":CALL IBWRT(SWEEPER%,WRT$)
1370 WRT$=STR$(F2):CALL IBWRT(SWEEPER%,WRT$)
1380 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1390 WRT$="PL":CALL IBWRT(SWEEPER%,WRT$)
1400 WRT$=STR$(P0):CALL IBWRT(SWEEPER%,WRT$)
1410 WRT$="DM":CALL IBWRT(SWEEPER%,WRT$)
1420 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
1430 WRT$=STR$(F1):CALL IBWRT(SWEEPER%,WRT$)
1440 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1450 WRT$="SS":CALL IBWRT(SWEEPER%,WRT$)
1460 WRT$=STR$(1000!*S0):CALL IBWRT(SWEEPER%,WRT$)
1470 WRT$="MZ":CALL IBWRT(SWEEPER%,WRT$)
1480 CLS
1485 N2=1
1490 REM SET UP GRAPH AXES
1500 LINE (25,10)-(25,210)
1510 LINE -(565,210)
1520 CALL IBCLR(SWEEPER%)
1530 WRT$="RF1":CALL IBWRT(SWEEPER%,WRT$)
1540 FOR I=1 TO N0
1550 F=F1+(I-1)*S0
1560 FOR J=1 TO 2
1580 A=J
1590 Q=M(J)*5/4095
1690 FOR V=1 TO N2
1700 NEXT V
1710 GOSUB 1890
1711 IF Y/M(J)>1.2 THEN 1713
1712 IF Y/M(J)>.099 THEN 1720
1713 GOSUB 2200
1714 GOTO 1580
1720 D(J)=Y
1780 YMAX=.9*(10^(P0/10)-1)
1785 REM PLOT DATA POINTS
1790 PSET (((F-F1)/(F2-F1))*540+25,210*(1-D(J)/YMAX))
1800 NEXT J
1810 WRITE #1,F*10^9,D(1)*10^-3,D(2)*10^-3
1820 CALL IBCLR(SWEEPER%)
1830 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
1835 WRT$="UP":CALL IBWRT(SWEEPER%,WRT$)
1840 NEXT I
1850 REM CALL IBCLR(SWEEPER%)
1860 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
1865 CALL IBLOC(SWEEPER%)
1870 WRITE #1,-999,-999,-999,-999
1880 CLOSE #1
1881 END
1890 REM CAMAC DATA FETCH
```

```
1900 REM CALL IBSIC(GPIB0%)
1910 REM WRT$="CAMAC":REM CALL IBFIND(WRT$,CAMAC%)
1920 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(0)
1930 CALL IBWRT(CAMAC%,WRT$)
1940 WRT$=CHR$(14)+CHR$(0)+CHR$(26)+CHR$(0)+CHR$(0)+CHR$(A)
1950 CALL IBWRT(CAMAC%,WRT$)
1960 WRT$=CHR$(14)+CHR$(0)+CHR$(16)+CHR$(0)+CHR$(0)+CHR$(A)
1970 CALL IBWRT(CAMAC%,WRT$)
1980 WRT$=CHR$(14)+CHR$(0)+CHR$(0)
1990 CALL IBWRT(CAMAC%,WRT$)
2000 CNT%=3:CALL IBRDI(CAMAC%,C%(0),CNT%)
2010 B1%=C%(0)/256
2020 Y=Q*(B1%*256+ABS(C%(1))+3)
2030 WRT$=CHR$(14)+CHR$(0)+CHR$(24)+CHR$(0)+CHR$(0)+CHR$(A)
2040 CALL IBWRT(CAMAC%,WRT$)
2050 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(96)
2060 CALL IBWRT(CAMAC%,WRT$)
2070 IF Y<0 THEN 1890
2080 RETURN
2090 REM METER ZERO
2100 CALL IBCLR(SWEEPER%)
2110 WRT$="F0":CALL IBWRT(SWEEPER%,WRT$)
2120 LOCATE 1,1
2130 PRINT "ENTER A Y WHEN METERS ZEROED":INPUT V$
2140 IF V$<>"Y" OR V$<>"y" THEN 2190
2150 WRT$="RF1":CALL IBWRT(SWEEPER%,WRT$)
2160 FOR V=1 TO 25
2170 NEXT V
2200 REM SCALE CHANGE
2210 LOCATE 1,1
2220 PRINT "SCALE SETTING FOR PORT ";J;" = ";M(J)
2230 PRINT "ENTER NEW SCALE VALUE":INPUT M(J)
2240 RETURN
```

ATTN.BAS

```
10        CLEAR    ,60000!            ' BASIC Declarations
20        IBINIT1 = 60000!
30        IBINIT2 = IBINIT1 + 3
40        BLOAD "bib.m",IBINIT1
50        CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,
                       IBPPC,IBBNA,IBONL,IBRSC,IBSRE,IBRSV,
                       IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,
                       IBEOT,IBRDF,IBWRTF)
60        CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,
                       IBCMD,IBCMDA,IBRD,IBRDA,IBSTOP,IBRPP,
                       IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,
                       IBWRTIA,IBSTA%,IBERR%,IBCNT%)
1000 CLS
1010 REM THE DETECTOR HOOK-UP IS AS FOLLOWS
1020 REM PORT 1 IS THE INCIDENT POWER MONITOR
1030 REM PORT 2 IS THE REFLECTED POWER MONITOR
1040 REM PORT 3 IS THE TRANSMITTED POWER MONITOR
1050 WRT$="SWEEPER":CALL IBFIND(WRT$,SWEEPER%)
1060 WRT$="CAMAC":CALL IBFIND(WRT$,CAMAC%)
1070 WRT$="GPIB0":CALL IBFIND(WRT$,GPIB0%)
1080 CALL IBCLR(SWEEPER%)
1085 REM TURN RF SOURCE OFF AND TURN ON CW FILTER
1086 REM AND INTERNAL LEVELING
1090 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
1092 WRT$="A1":CALL IBWRT(SWEEPER%,WRT$)
1094 WRT$="FL1":CALL IBWRT(SWEEPER%,WRT$)
1100 REM HEADER BLOCK ENTRY
1110 PRINT "ENTER THE FILENAME"
1120 INPUT V$
1130 OPEN "O",#1,V$
1140 PRINT "ENTER TEST LABEL (IN SINGLE QUOTES)":INPUT V$
1150 I0=0
1160 PRINT "ENTER SAMPLE ID#":INPUT L1
1170 PRINT "ENTER ID FOR INC DIR COUPLER":INPUT D1
1180 PRINT "ENTER ID FOR REF DIR COUPLER":INPUT D2
1190 PRINT "ENTER ID FOR INC THERM MOUNT":INPUT T1
1200 PRINT "ENTER ID FOR REF THERM MOUNT":INPUT T2
1210 PRINT "ENTER ID FOR TRANS THERM MOUNT":INPUT T3
1220 REM SEND HEADER TO 11/60
1230 PRINT #1,V$
1240 WRITE #1,I0
1250 PRINT #1,"'CABLE ID'"
1260 WRITE #1,L1
1270 PRINT #1,"'INC DIR COUPLER ID'"
1280 WRITE #1,D1
1290 PRINT #1,"'REF DIR COUPLER ID'"
1300 WRITE #1,D2
1310 PRINT #1,"'INC THERM ID'"
1320 WRITE #1,T1
1330 PRINT #1,"'REF THERM ID'"
1340 WRITE #1,T2
1350 PRINT #1,"'TRANS THERM MOUNT ID'"
```

```
1360 WRITE #1,T3
1370 PRINT #1,''FREQ(HZ) INC POW(W)  REF POW(W)  TRANS POW(W)''
1375 REM SET CALIBRATION PARAMETERS
1380 F1=1!
1390 F2=18!
1400 P0=13
1405 REM NUMBER OF FREQUENCY STEPS
1410 N0=200
1430 C0=1
1450 DIM D(3),M(3)
1455 REM SET FREQUENCY STEP SIZE
1460 S0=(F2-F1)/(N0-1)
1470 CALL IBCLR(SWEEPER%)
1480 WRT$="FA":CALL IBWRT(SWEEPER%,WRT$)
1490 WRT$=STR$(F1): CALL IBWRT(SWEEPER%,WRT$)
1500 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1510 WRT$="FB":CALL IBWRT(SWEEPER%,WRT$)
1520 WRT$=STR$(F2): CALL IBWRT(SWEEPER%,WRT$)
1530 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1540 WRT$="PL":CALL IBWRT(SWEEPER%,WRT$)
1550 WRT$=STR$(P0): CALL IBWRT(SWEEPER%,WRT$)
1560 WRT$="DM":CALL IBWRT(SWEEPER%,WRT$)
1562 WRT$="SM": CALL IBWRT(SWEEPER%,WRT$)
1563 WRT$=STR$(F1): CALL IBWRT(SWEEPER%,WRT$)
1564 WRT$="GZ": CALL IBWRT(SWEEPER%,WRT$)
1565 WRT$="SS": CALL IBWRT(SWEEPER%,WRT$)
1566 WRT$=STR$(S0*1000): CALL IBWRT(SWEEPER%,WRT$)
1567 WRT$="MZ": CALL IBWRT(SWEEPER%,WRT$)
1570 PRINT "ENTER THE SCALE FACTORS FOR THE POWER METERS"
1580 PRINT "SCALE FACTOR FOR PORT 1":INPUT M(1)
1590 PRINT "SCALE FACTOR FOR PORT 2":INPUT M(2)
1600 PRINT "SCALE FACTOR FOR PORT 3":INPUT M(3)
1610 CLS
1615 REM DRAW AXES
1620 LINE (25,10)-(25,210)
1630 LINE -(565,210)
1640 CALL IBCLR(SWEEPER%)
1650 WRT$="RF1":CALL IBWRT(SWEEPER%,WRT$)
1660 FOR I=1 TO N0
1670 F=F1+(I-1)*S0
1680 FOR K=1 TO 3
1700 A=K
1710 Q=M(K)*5/4095
1810 FOR V=1 TO C0
1820 NEXT V
1830 GOSUB 2060
1840 IF Y/M(K)>1.2 THEN 1860
1850 IF Y/M(K)>.099 THEN 1880
1860 GOSUB 2260
1870 GOTO 1700
1880 D(K)=Y
1940 YMAX=(10^(P0/10))-1
1945 REM PLOT DATA POINT ON GRAPH
1950 PSET (((F-F1)/(F2-F1))*540+25,210*(1-D(K)/YMAX))
1960 NEXT K
```

```
1970 WRITE #1,F*10^9,D(1)*10^-3,D(2)*10^-3,D(3)*10^-3
1980 CALL IBCLR(SWEEPER%)
1990 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
2000 WRT$="UP":CALL IBWRT(SWEEPER%,WRT$)
2010 NEXT I
2030 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
2035 CALL IBLOC(SWEEPER%)
2040 WRITE #1,-999,-999,-999,-999
2050 CLOSE #1
2051 END
2060 REM CAMAC DATA FETCH
2070 REM CALL IBSIC(GPIB0%)
2080 REM WRT$="CAMAC":REM CALL IBFIND(WRT$,CAMAC%)
2090 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(0)
2100 CALL IBWRT(CAMAC%,WRT$)
2110 WRT$=CHR$(14)+CHR$(0)+CHR$(26)+CHR$(0)+CHR$(0)+CHR$(A)
2120 CALL IBWRT(CAMAC%,WRT$)
2130 WRT$=CHR$(14)+CHR$(0)+CHR$(16)+CHR$(0)+CHR$(0)+CHR$(A)
2140 CALL IBWRT(CAMAC%,WRT$)
2150 WRT$=CHR$(14)+CHR$(0)+CHR$(0)
2160 CALL IBWRT(CAMAC%,WRT$)
2170 CNT%=3:CALL IBRDI(CAMAC%,C%(0),CNT%)
2180 B1%=C%(0)/256
2190 Y=Q*(B1%*256+ABS(C%(1))+3)
2200 WRT$=CHR$(14)+CHR$(0)+CHR$(24)+CHR$(0)+CHR$(0)+CHR$(A)
2210 CALL IBWRT(CAMAC%,WRT$)
2220 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(96)
2230 CALL IBWRT(CAMAC%,WRT$)
2240 IF Y<0 THEN 2060
2250 RETURN
2260 REM SCALE CHANGE
2270 LOCATE 1,1
2280 PRINT "SCALE SETTING FOR PORT";K;" = ";M(K)
2290 PRINT "ENTER NEW SCALE VALUE":INPUT M(K)
2300 RETURN
2310 REM METER ZERO
2320 CALL IBCLR(SWEEPER%)
2330 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
2340 LOCATE 1,1
2350 PRINT "ENTER A Y WHEN METERS ZEROED":INPUT V$
2360 IF V$<>"Y" OR V$<>"y" THEN 2410
2370 WRT$="RF1":CALL IBWRT(SWEEPER%,WRT$)
2380 FOR V=1 TO 25
2390 NEXT V
2460 RETURN
```

XTAL.BAS

```
10        CLEAR    ,60000!          ' BASIC Declarations
20        IBINIT1 = 60000!
30        IBINIT2 = IBINIT1 + 3
40        BLOAD "bib.m",IBINIT1
50        CALL IBINIT1(IBFIND,IBTRG,IBCLR,IBPCT,IBSIC,IBLOC,
                       IBPPC,IBBNA,IBONL,IBRSC,IBSRE,IBRSV,
                       IBPAD,IBSAD,IBIST,IBDMA,IBEOS,IBTMO,
                       IBEOT,IBRDF,IBWRTF)
60        CALL IBINIT2(IBGTS,IBCAC,IBWAIT,IBPOKE,IBWRT,IBWRTA,
                       IBCMD,IBCMDA,IBRD,IBRDA,IBSTOP,IBRPP,
                       IBRSP,IBDIAG,IBXTRC,IBRDI,IBWRTI,IBRDIA,
                       IBWRTIA,IBSTA%,IBERR%,IBCNT%)
1000 REM THIS PROGRAM RUNS A DOUBLE FOR-NEXT LOOP
1010 REM THE OUTER LOOP IS A POWER SWEEP
1020 REM THE INNER LOOP IS A FREQ SWEEP
1030 CLS
1040 BD$="SWEEPER":CALL IBFIND(BD$,SWEEPER%)
1050 BD$="GPIB0":CALL IBFIND(BD$,GPIB0)
1060 CALL IBCLR(SWEEPER%)
1065 REM TURN RF OFF AND TURN ON INTERNAL LEVELING
1066 REM AND CW FILTER
1070 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
1080 WRT$="A1":CALL IBWRT(SWEEPER%,WRT$)
1090 WRT$="FL1":CALL IBWRT(SWEEPER%,WRT$)
1100 PRINT "ENTER THE FILE NAME"
1110 INPUT V$
1120 OPEN "O",#1,V$
1130 PRINT "ENTER THE TEST LABEL(IN SINGLE QUOTES)"
1140 INPUT V$
1150 PRINT #1,V$
1160 REM SEND DATA FLAG
1170 WRITE #1,0
1180 REM HEADER BLOCK
1190 DIM D(3),M(3),T(2),C2%(3)
1200 PRINT "ENTER XTAL ID#":INPUT D(1)
1210 PRINT "ENTER THE INC DIRECTIONAL COUPLER ID#":INPUT D(2)
1220 PRINT "ENTER THE REF DIRECTIONAL COUPLER ID#":INPUT D(3)
1230 PRINT "ENTER THE INC THERM MOUNT ID#":INPUT T(1)
1240 PRINT "ENTER THE REF THERM MOUNT ID#":INPUT T(2)
1250 PRINT #1,""XTAL ID#""
1260 WRITE #1,D(1)
1270 PRINT #1,""INC DRC ID#""
1280 WRITE #1,D(2)
1290 PRINT #1,""REF DRC ID#""
1300 WRITE #1,D(3)
1310 PRINT #1,""INC THERM ID#""
1320 WRITE #1,T(1)
1330 PRINT #1,""REF THERM ID#""
1340 WRITE #1,T(2)
1350 REM POWER METER SCALE FACTORS
1380 M(3)=1
1390 PRINT "ENTER A Y IF THE XTAL GIVES A POSITIVE "
```

```
1391 PRINT "SIGNAL INTO 50 OHMS"
1395 PRINT "MAKE SURE THE GAIN VALUE FOR THE"
1396 PRINT " XTAL IS SET PROPERLY"
1397 PRINT "IN LINE 1580"
1400 PRINT "THE XTAL IS ON PORT 3"
1420 PRINT "INC. POWER ON PORT 1"
1430 PRINT "REF. POWER ON PORT 2"
1440 INPUT V$
1450 IF V$=CHR$(89) OR V$=CHR$(121) THEN 1460 ELSE 1440
1460 PRINT "ENTER THE SCALE FACTORS FOR THE INC AND "
1461 PRINT "REF POWER METERS"
1470 INPUT M(1)
1480 INPUT M(2)
1490 REM SETUP FREQUENCY SWEEPER
1495 REM POWER LEVEL RANGE IN DBM
1500 P1=0
1510 P2=10!
1515 REM FREQUENCY RANGE IN GHZ
1520 F1=.5
1530 F2=12.5
1535 REM NUMBER OF FREQUENCY STEPS < 514
1540 N0=100
1545 REM NUMBER OF POWER LEVEL STEPS < 10
1550 N1=8
1560 REM MAX VOLTAGE LEVEL OUT OF CRYSTALS
1570 YMAX=.1
1575 REM GAIN VALUE FROM AMPLIFIER BOX
1580 G1=58!
1585 REM STEP SIZES FOR LOOPS
1590 S0=(F2-F1)/(N0-1)
1600 S1=(P2-P1)/(N1-1)
1605 REM SEND COMMANDS TO SWEEP OSCILLATOR
1610 WRT$="FA":CALL IBWRT(SWEEPER%,WRT$)
1620 WRT$=STR$(F1):CALL IBWRT(SWEEPER%,WRT$)
1630 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1640 WRT$="FB":CALL IBWRT(SWEEPER%,WRT$)
1650 WRT$=STR$(F2):CALL IBWRT(SWEEPER%,WRT$)
1660 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1670 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
1680 WRT$=STR$(F1):CALL IBWRT(SWEEPER%,WRT$)
1690 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1700 WRT$="SS":CALL IBWRT(SWEEPER%,WRT$)
1710 WRT$=STR$(S0*1000!):CALL IBWRT(SWEEPER%,WRT$)
1720 WRT$="MZ":CALL IBWRT(SWEEPER%,WRT$)
1730 WRT$="PL":CALL IBWRT(SWEEPER%,WRT$)
1740 WRT$=STR$(P1):CALL IBWRT(SWEEPER%,WRT$)
1750 WRT$="DM":CALL IBWRT(SWEEPER%,WRT$)
1760 WRT$="SS":CALL IBWRT(SWEEPER%,WRT$)
1770 WRT$=STR$(S1):CALL IBWRT(SWEEPER%,WRT$)
1780 WRT$="DM":CALL IBWRT(SWEEPER%,WRT$)
I790 REM TAKE DATA
1800 CLS
1805 REM SET UP GRAPH AXES
1810 LINE (25,10)-(25,210)
1820 LINE -(565,210)
```

```
1825 REM TURN ON RF SOURCE
1830 WRT$="RF1":CALL IBWRT(SWEEPER%,WRT$)
1840 FOR P=P1 TO P2 STEP S1
1850 PRINT #1,""POWER LEVEL (DBM)""
1860 WRITE #1,P
1870 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
1880 WRT$=STR$(F1):CALL IBWRT(SWEEPER%,WRT$)
1890 WRT$="GZ":CALL IBWRT(SWEEPER%,WRT$)
1900 PRINT #1,""F(HZ)       INC(W)        REF(W)        XTAL(V)""
1910 FOR F=F1 TO F2 STEP S0
1920 FOR I=1 TO 3
1940 A=I
1950 Q=M(I)*5/4095
2040 REM GOSUB 2320
2050 GOSUB 2360
2060 D(I)=Y
2065 REM IGNORE VOLTAGE LEVELS FOR XTAL AND REFLECTED POWER
2070 IF I=2 OR I=3 THEN 2180
2080 IF D(I)/M(I)>1.2 THEN 2100
2090 IF D(I)/M(I)>.099 THEN 2180
2100 GOSUB 2710
2110 GOTO 1940
2180 NEXT I
2185 REM PLOT VOLTAGE VARIATION OF XTAL OUTPUT
2190 PSET (((F-F1)/(F2-F1))*540+25,210*(1-(D(3)/G1)/YMAX))
2200 WRITE #1,F*10^9,D(1)*10^-3,D(2)*10^-3,D(3)/G1
2210 WRT$="SM":CALL IBWRT(SWEEPER%,WRT$)
2220 WRT$="UP":CALL IBWRT(SWEEPER%,WRT$)
2230 NEXT F
2240 WRITE #1,-999,-999,-999,-999
2250 WRT$="PL":CALL IBWRT(SWEEPER%,WRT$)
2260 WRT$="UP":CALL IBWRT(SWEEPER%,WRT$)
2270 NEXT P
2280 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
2290 CALL IBLOC(SWEEPER%)
2300 CLOSE #1
2310 END
2320 REM DELAY SUBROUTINE ****************************************
2330 FOR J=1 TO 10
2340 NEXT J
2350 RETURN
2360 REM CAMAC DATA FETCH ****************************************
2370 CALL IBSIC(GPIB0%)
2380 WRT$="CAMAC":CALL IBFIND(WRT$,CAMAC%)
2390 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(0)
2400 CALL IBWRT(CAMAC%,WRT$)
2410 WRT$=CHR$(14)+CHR$(0)+CHR$(26)+CHR$(0)+CHR$(0)+CHR$(A)
2420 CALL IBWRT(CAMAC%,WRT$)
2430 WRT$=CHR$(14)+CHR$(0)+CHR$(16)+CHR$(0)+CHR$(0)+CHR$(A)
2440 CALL IBWRT(CAMAC%,WRT$)
2480 WRT$=CHR$(14)+CHR$(0)+CHR$(0)
2490 CALL IBWRT(CAMAC%,WRT$)
2500 CNT%=3:CALL IBRDI(CAMAC%,C2%(0),CNT%)
2510 B1%=C2%(0)/256
2520 Y=Q*(B1%*256+ABS(C2%(1))+3)
```

```
2540 WRT$=CHR$(14)+CHR$(0)+CHR$(24)+CHR$(0)+CHR$(0)+CHR$(A)
2550 CALL IBWRT(CAMAC%,WRT$)
2560 WRT$=CHR$(30)+CHR$(0)+CHR$(17)+CHR$(0)+CHR$(0)+CHR$(96)
2570 CALL IBWRT(CAMAC%,WRT$)
2600 RETURN
2610 REM METER ZERO ***********************************************
2630 WRT$="RF0":CALL IBWRT(SWEEPER%,WRT$)
2640 LOCATE 1,1
2650 PRINT "ENTER A Y WHEN FINISHED ZEROING METERS"
2660 INPUT V$
2670 IF V$=CHR$(89) OR V$=CHR$(121) THEN 2690 ELSE 2640
2680 IF V$=CHR$(89) OR V$=CHR$(121) THEN 2700 ELSE 2650
2690 GOSUB 2320
2700 RETURN
2710 REM SCALE CHANGE
2720 IF I=3 THEN 2800
2730 K=I
2740 LOCATE 1,1
2750 PRINT "
2760 LOCATE 1,1
2770 PRINT "SCALE SETTING FOR PORT";I;"=";M(I)
2780 PRINT "ENTER NEW VALUE"
2790 INPUT M(I)
2800 RETURN
```

DRC.FOR

```
C THIS PROGRAM WILL TAKE THE RAW DATA OF A DIRECTIONAL COUPLER
C CALIBRATION RUN AND PLOT OUT THE ATTENUATION VS FREQUENCY.
C IF THIS PLOT IS INCORRECT THE INPUT DATA FILE SHOULD BE REPEATED.
      DIMENSION FREQ(514),PIN(514),POUT(514),ATTN(514),TEMP(4)
      DIMENSION POW(514)
      CHARACTER*70 TEXT1,TEXT2,TEXT3,TEXT4,TEXT5
      CHARACTER  FILE*40,NAME*10
      CHARACTER*20 PTITLE,XLAB,YLAB
      INTEGER THMIN,THMOUT,DRC
C GET DATA FILE OPENED AND READY TO READ
      WRITE(*,*) 'ENTER THE FILE NAME TO USE(IN SINGLE QUOTES)'
      READ(*,*) FILE
      OPEN(UNIT=1,NAME=FILE,STATUS='OLD')
      READ(1,*) IFL
      READ(1,*)TEXT1,TEXT5,IPO
      READ(1,*) TEXT2
      READ(1,*) DRC
      READ(1,*) TEXT2
      READ(1,*) THMIN
      READ(1,*) TEXT3
      READ(1,*) THMOUT
      READ(1,*) TEXT4
      I=1
10    READ(1,*) FREQ(I),PIN(I),POUT(I)
      IF ((FREQ(I).EQ.-999).OR.(PIN(I).EQ.-999).OR.(POUT(I).EQ.-999))
     +GOTO 14
      I=I+1
      GOTO 10
14    I=I-1
      CLOSE(UNIT=1)
C ALL THE DATA IS READ IN NOW USE IT
      IF (I.GT.512) THEN
         WRITE(*,*) 'THERE ARE TO MANY DATA POINTS'
      ELSE
C CHECK FOR OUTPUT OF RAW DATA OR REDUCED DATA
      WRITE(*,*) 'ENTER A 1 FOR RAW DATA OR A 2 FOR REDUCED DATA'
      READ(*,*) K
      IF (K.EQ.1) THEN
C WE WILL CORRECT OUT THE THERMISTOR MOUNT FREQUENCY RESPONSE
C THE UNCORRECTED DATA IS FEED TO SUBROUTINE TCOR  CORRECTED
C DATA IS RETURNED IN TH SAME ARRAYS.
C CORRECT PIN
      DO 15 J=1,I
         POW(J)=PIN(J)
15    CONTINUE
      CALL TCOR(I,FREQ,POW,THMIN)
      DO 16 J=1,I
         PIN(J)=POW(J)
16    CONTINUE
C CORRECT POUT
      DO 17 J=1,I
         POW(J)=POUT(J)
```

```
17     CONTINUE
       CALL TCOR(I,FREQ,POW,THMOUT)
       DO 18 J=1,I
          POUT(J)=POW(J)
18     CONTINUE
C PLOT RAW INCIDENT AND PICKOFF POWER ON THE SAME SCALE
C USE VAX GRAPHICS
C GRAPHICS FOR USE ON 4010,4014,4025,4105
       WRITE(*,*) 'ENTER THE DEVICE ID#'
       READ(*,*) IDEVIC
       IF (IDEVIC.EQ.4014.OR.IDEVIC.EQ.4105) THEN
          XMI=.01
          XMA=.9
          YMI=.01
          YMA=.9
       ELSEIF (IDEVIC.EQ.4010) THEN
          XMI=.01
          XMA=.6
          YMI=.01
          YMA=.6
       ELSEIF (IDEVIC.EQ.4025) THEN
          XMI=.05
          XMA=.85
          YMI=.05
          YMA=.85
          IDEVIC=4010
       ELSE
          WRITE(*,*) 'DEVICE ',IDEVIC,' NOT USED'
          STOP
       ENDIF
       CALL GSTART(IDEVIC)
       CALL NEWPAG
       CALL MINMAX(FREQ,I,XWMIN,XWMAX)
       CALL MINMAX(PIN,I,YMIN,YMAX)
       CALL MINMAX(POUT,I,YM,YM1)
       IF (YMIN.LE.YM) YWMIN=YMIN
       IF (YMIN.GT.YM) YWMIN=YM
       IF (YMAX.GE.YM1) YWMAX=YMAX
       IF (YMAX.LT.YM1) YWMAX=YM1
       CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
       CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
       CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
       XLAB='FREQ(HZ)'
       YLAB='POW(W)'
       PTITLE='RAW DATA'
       CALL XLABEL(XLAB)
       CALL YLABEL(YLAB)
       CALL TITLE(PTITLE)
       CALL CURVE(FREQ,PIN,I,0,+1)
       CALL CURVE(FREQ,POUT,I,0,+1)
       CALL GRSTOP
C COMPUTE ATTENUATION FOR UNCORRECTED DATA IN DB
       ELSEIF (K.EQ.2) THEN
C REMOVE THEMSTOR CAL FACTORS
       DO 21 J=1,I
```

```
         POW(J)=PIN(J)
21     CONTINUE
       CALL TCOR(I,FREQ,POW,THMIN)
       DO 22 J=1,I
         PIN(J)=POW(J)
22     CONTINUE
       DO 23 J=1,I
         POW(J)=POUT(J)
23     CONTINUE
       CALL TCOR(I,FREQ,POUT,THMOUT)
       DO 24 J=1,I
         POUT(J)=POW(J)
24     CONTINUE
       DO 30 J=1,I
         ATTN(J)=10.*ALOG10(PIN(J)/POUT(J))
30     CONTINUE
C PLOT ATTN VS FREQ.
C USE VAX GRAPHICS
C GRAPHICS FOR USE ON 4010,4014,4025,4105
       WRITE(*,*) 'ENTER THE DEVICE ID#'
       READ(*,*) IDEVIC
       IF (IDEVIC.EQ.4014.OR.IDEVIC.EQ.4105) THEN
         XMI=.01
         XMA=.9
         YMI=.01
         YMA=.9
       ELSEIF (IDEVIC.EQ.4010) THEN
         XMI=.01
         XMA=.6
         YMI=.01
         YMA=.6
       ELSEIF (IDEVIC.EQ.4025) THEN
         XMI=.05
         XMA=.85
         YMI=.05
         YMA=.85
         IDEVIC=4010
       ELSE
         WRITE(*,*) 'DEVICE ',IDEVIC,' NOT USED'
         STOP
       ENDIF
       CALL GSTART(IDEVIC)
       CALL NEWPAG
       CALL MINMAX(FREQ,I,XWMIN,XWMAX)
       CALL MINMAX(ATTN,I,YWMIN,YWMAX)
       CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
       CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
       CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
       XLAB='FREQ(HZ)'
       YLAB='ATTN(DB)'
       PTITLE='FREQ. RESPONSE'
       CALL XLABEL(XLAB)
       CALL YLABEL(YLAB)
       CALL TITLE(PTITLE)
       CALL CURVE(FREQ,ATTN,I,0,+1)
```

```
      CALL GRSTOP
C CREATE FILE FOR LOOKING AT ATTN VALUES
      OPEN(UNIT=1,FILE='DRATTN.PAT',STATUS='NEW')
      WRITE(1,*) '''FLAG SET TO 1 MEANS DATA CORRECTED'''
      WRITE(1,*) 1
      WRITE(1,*) '''DIR. COUPLER ID#=  '''
      WRITE(1,*) DRC
      WRITE(1,*) I
      WRITE(1,40)
40    FORMAT(' ',''FREQ(HZ)',20X,'ATTN(DB)''')
      DO 50 J=1,I
         WRITE(1,45) FREQ(J),ATTN(J)
45       FORMAT(' ',1PE11.3,17X,1PE11.3)
50    CONTINUE
      CLOSE(UNIT=1)
      ENDIF
      ENDIF
      STOP
      END
C
C THIS SUBROUTINE TCOR
      SUBROUTINE TCOR(J,FREQ,POW,ITHMIN)
      DIMENSION FREQ(J),POW(J)
      REAL*8 NU(50),CFAC(50),B(50),C(50),D(50),FRQ,CAL
      CHARACTER FNAME*20
C GET THE DATA FILE  THERMISTOR MOUNT ITHMIN
      IF (ITHMIN.EQ.21593) FNAME='21593.MNT'
      IF (ITHMIN.EQ.21594) FNAME='21594.MNT'
      IF (ITHMIN.EQ.18832) FNAME='18832.MNT'
      IF (ITHMIN.EQ.21627) FNAME='21627.MNT'
      IF (ITHMIN.EQ.51009) FNAME='51009.MNT'
      IF (ITHMIN.EQ.50918) FNAME='50918.MNT'
      IF (ITHMIN.EQ.19131) FNAME='19131.MNT'
      OPEN(UNIT=2,FILE=FNAME,STATUS='OLD')
C GET THE NUMBER OF DATA PAIR
      READ(2,*) MAX
      DO 10 I=1,MAX
         READ(2,2) NU(I),CFAC(I)
2     FORMAT(2D10.2)
10    CONTINUE
      CLOSE(UNIT=2)
C GET THE INTERPOLATION ARRAYS FILLED
      CALL SPCOEF(MAX,NU,CFAC,B,C,D,IFLAG)
      IF (IFLAG.NE.0) THEN
         WRITE(*,*) 'ERROR IN INTERPOLATON, ',IFLAG
      ENDIF
C CORRECT ARRAY POW
C ARRAY NU IS IN GHZ, AND ARRAY FREQ IS IN HZ SO WE MUST TAKE
C HE DIFFERENT UNITS
      DO 20 I=1,J
         FRQ=DBLE(FREQ(I))*(1.D-9)
         CALL SVALUE(MAX,NU,CFAC,B,C,D,FRQ,CAL,IFLAG)
         IF (IFLAG.NE.0) THEN
            WRITE(*,*) 'ERROR IN EVALUATION, ',IFLAG,FREQ(I)
        ENDIF
```

```
        POW(I)=(POW(I))/CAL
20     CONTINUE
C THE ARRAY POW IS NOW CORRECTED FOR THE THERMISTOR MOUNT USED
       RETURN
       END
```

DB.FOR

```
      PROGRAM DB
C THIS PROGRAM WILL READ A FILE GENERATED BY THE ATTN PROGRAM ON THE
C 4051 CALIBRATION SYSTEM
      DIMENSION FREQ(514),POW(514,3),ITHERM(3),IDRC(2),TEMP(514)
      CHARACTER LABEL1*20,LABEL2*50,XLAB*15,YLAB*15,PTITLE*15
      CHARACTER FNAME*40
C GET THE DATA FILE NAME TO LOOK AT
      WRITE(*,*) 'ENTER THE DATA FILE NAME'
      READ(*,*) FNAME
      OPEN(UNIT=1,FILE=FNAME,STATUS='OLD')
      READ(1,*) LABEL1
      READ(1,*) IFLAG
      READ(1,*) LABEL1
      READ(1,*) ISMPLE
      READ(1,*) LABEL1
      READ(1,*) IDRC(1)
      READ(1,*) LABEL1
      READ(1,*) IDRC(2)
      READ(1,*) LABEL1
      READ(1,*) ITHERM(1)
      READ(1,*) LABEL1
      READ(1,*) ITHERM(2)
      READ(1,*) LABEL1
      READ(1,*) ITHERM(3)
      READ(1,*) LABEL2
      I=1
5     READ(1,*) FREQ(I),POW(I,1),POW(I,2),POW(I,3)
      IF(FREQ(I).EQ.-999.OR.POW(I,1).EQ.-999.OR.POW(I,2).EQ.-999.OR.
     +POW(I,3).EQ.-999) GOTO 10
      I=I+1
      GOTO 5
10    I=I-1
      CLOSE(UNIT=1)
      IF (I.GT.512) THEN
         WRITE(*,*) 'TO MANY DATA POINTS'
         STOP
      ENDIF
C REMOVE THERMISTOR RESPONSE FROM POW(I,J) FOR J=1,2,3
      DO 15 K=1,3
         DO 12 L=1,I
            TEMP(L)=POW(L,K)
12       CONTINUE
         ITH=ITHERM(K)
         CALL TCOR(I,FREQ,TEMP,ITH)
         DO 14 L=1,I
            POW(L,K)=TEMP(L)
14       CONTINUE
15    CONTINUE
C REMOVE THE DIRECTIONAL COUPLER RESPONSE FROM POW(L,K) FOR K=1,2
      DO 25 K=1,2
         DO 20 L=1,I
            TEMP(L)=POW(L,K)
```

```
20        CONTINUE
          ID=IDRC(K)
          CALL DCCOR(I,FREQ,TEMP,ID)
          DO 24 L=1,I
             POW(L,K)=TEMP(L)
24        CONTINUE
25     CONTINUE
C NOW THE POWER VALUES HAVE BEEN CORRECTED FOR THE FREQUENCY RESPONSE
C OF THE THERMISTOR MOUNTS AND THE DIRECTIONAL COUPLERS USED TO TAKE
C THE VARIOUS POWER VALUES.
C COMP ATTENUATION, ATTN=(POW(L,1)-POW(L,2))/POW(L,3)
29     WRITE(*,*) 'ENTER A 1 IF YOU WANT THE ATTENUATION COMPUTED'
       WRITE(*,*) 'WITH ONLY THE INCIDENT POWER'
       WRITE(*,*) 'ENTER A 0 IF YOU WANT THE ATTENUATION COMPUTED'
       WRITE(*,*) 'USING THE NET POWER'
       READ(*,*) IOPT
       IF (IOPT.NE.1.AND.IOPT.NE.0) GOTO 29
       DO 30 L=1,I
          IF (IOPT.EQ.0) THEN
             ATTN=(POW(L,1)-POW(L,2))/POW(L,3)
C TEMP(L)=0. MEANS THEIR IS AN ERROR IN THE DATA
             IF (ATTN.LE.1.) THEN
                TEMP(L)=0.
             ELSE
                TEMP(L)=10.*ALOG10(ATTN)
             ENDIF
          ELSE
             ATTN=POW(L,1)/POW(L,3)
             IF (ATTN.LE.1) THEN
                TEMP(L)=0.
             ELSE
                TEMP(L)=10.*ALOG10(ATTN)
             ENDIF
          ENDIF
30     CONTINUE
C MAKE A GRAPH OF THE FREQUENCY RESPONSE
       XLAB='FREQ(HZ)'
       YLAB='ATTN(DB)'
C USE VAX GRAPHICS
C GRAPHICS FOR USE ON 4010,4014,4025,4105
       WRITE(*,*) 'ENTER THE DEVICE ID#'
       READ(*,*) IDEVIC
       IF (IDEVIC.EQ.4014.OR.IDEVIC.EQ.4105) THEN
          XMI=.01
          XMA=.9
          YMI=.01
          YMA=.9
       ELSEIF (IDEVIC.EQ.4010) THEN
          XMI=.01
          XMA=.6
          YMI=.01
          YMA=.6
       ELSEIF (IDEVIC.EQ.4025) THEN
          XMI=.05
          XMA=.85
```

```
         YMI=.05
         YMA=.85
         IDEVIC=4010
      ELSE
         WRITE(*,*) 'DEVICE ',IDEVIC,' NOT USED'
         STOP
      ENDIF
      CALL GSTART(IDEVIC)
      CALL NEWPAG
      CALL MINMAX(FREQ,I,XWMIN,XWMAX)
      CALL MINMAX(TEMP,I,YWMIN,YWMAX)
      CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
      CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
      CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
      IF (IOPT.EQ.0) PTITLE='NET POWER'
      IF (IOPT.EQ.1) PTITLE='INC. POWER'
      CALL XLABEL(XLAB)
      CALL YLABEL(YLAB)
      CALL TITLE(PTITLE)
      CALL CURVE(FREQ,TEMP,I,0,+1)
      CALL GRSTOP
C COMPUTE ATTENUATION FOR UNCORRECTED DATA IN DB
C CREATE FILE FOR SAVING DATA
      OPEN(UNIT=1,FILE='SAMPLE.DAT',STATUS='NEW')
      WRITE(1,*) '''FLAG SET TO 1 MEANS DATA IS CORRECTED'''
      WRITE(1,*) 1
      WRITE(1,*) '''SAMPLE ID TAG'''
      WRITE(1,*) ISMPLE
      WRITE(1,*) I
      WRITE(1,40)
40    FORMAT(' ',''''FREQ(HZ)',19X,'ATTN(DB)''')
      DO 50 L=1,I
         WRITE(1,45) FREQ(L),TEMP(L)
45       FORMAT(' ',1PE11.3,17X,1PE11.3)
50    CONTINUE
      CLOSE(UNIT=1)
      STOP
      END
C
C SUBROUTINE TCOR
      SUBROUTINE TCOR(J,FREQ,TEMP,ITHERM)
      DIMENSION FREQ(J),TEMP(J)
      REAL*8 NU(50),CFAC(50),B(50),C(50),D(50),FRQ,CAL
      CHARACTER*20 FNAME
      IF (ITHERM.EQ.21593) FNAME='21593.MNT'
      IF (ITHERM.EQ.21594) FNAME='21594.MNT'
      IF (ITHERM.EQ.18832) FNAME='18832.MNT'
      IF (ITHERM.EQ.21627) FNAME='21627.MNT'
      IF (ITHERM.EQ.51009) FNAME='51009.MNT'
      IF (ITHERM.EQ.50918) FNAME='50918.MNT'
      IF (ITHERM.EQ.19131) FNAME='19131.MNT'
      OPEN(UNIT=2,FILE=FNAME,STATUS='OLD')
C GET THE NUMBER OF DATA POINTS
      READ(2,*) MAX
      DO 10 I=1,MAX
```

```
         READ(2,2) NU(I),CFAC(I)
2        FORMAT(2D10.2)
10    CONTINUE
C CORRECT ARRAY TEMP BY INTERPOLATING ARRAY CFAC
      CALL SPCOEF(MAX,NU,CFAC,B,C,D,IFLAG)
      IF (IFLAG.NE.0) THEN
         WRITE(*,*)'ERROR IN INTERPOLATION ',IFLAG
      ENDIF
      DO 20 I=1,J
         FRQ=DBLE(FREQ(I))*(1.D-9)
         CALL SVALUE(MAX,NU,CFAC,B,C,D,FRQ,CAL,IFLAG)
         IF (IFLAG.NE.0) THEN
            WRITE(*,*)'ERROR IN EVALUATION ',IFLAG,FREQ(I)
         ENDIF
         TEMP(I)=TEMP(I)/CAL
20    CONTINUE
      RETURN
      END
C
      SUBROUTINE DCCOR(J,FREQ,TEMP,ID)
      DIMENSION FREQ(J),TEMP(J)
      REAL*8 NU(514),PCKOFF(514),B(514),C(514),D(514),FRQ,VAL
      CHARACTER FNAME*20,LABEL*50
      IF (ID.EQ.1) FNAME='DC1.DAT'
      IF (ID.EQ.2) FNAME='DC2.DAT'
      IF (ID.EQ.3) FNAME='DC3.DAT'
      IF (ID.EQ.4) FNAME='DC4.DAT'
      OPEN(UNIT=2,FILE=FNAME,STATUS='OLD')
      READ(2,*) LABEL
      READ(2,*) I
      READ(2,*) LABEL
      READ(2,*) I
      READ(2,*) MAX
      READ(2,*) LABEL
      DO 5 I=1,MAX
         READ(2,*) A,Z
         NU(I)=DBLE(A)
         PCKOFF(I)=DBLE(Z)
5     CONTINUE
C INTERPOLATE THE DATA
      CALL SPCOEF(MAX,NU,PCKOFF,B,C,D,IFLAG)
      IF (IFLAG.NE.0) THEN
         WRITE(*,*) 'ERROR IN INTERPOLATION',IFLAG
      ENDIF
C REMOVE DIR. COUPLERSE
      DO 10 I=1,J
         FRQ=DBLE(FREQ(I))
         CALL SVALUE(MAX,NU,PCKOFF,B,C,D,FRQ,VAL,IFLAG)
         IF (IFLAG.NE.0) THEN
            WRITE(*,*)'ERROR IN PICKOFF EVALUATION',IFLAG,FREQ(I)
         ENDIF
         TEMP(I)=VAL*TEMP(I)
10    CONTINUE
      RETURN
      END
```

XTAL.FOR

```
      PROGRAM XTAL
C THIS PROGRAM WILL READ A FILE GENERATED BY THE XTAL PROGRAM ON THE
C 4051 CALIBRATION SYSTEM
C FREQ(I,J) HOLDS THE FREQUENCY
C POWIN(I,J) HOLDS THE INCIENT PICKOFF POWER
C POWREF(I,J) HOLDS THE REFLECTED PICKOFF POWER
C VOLT(I,J) HOLDS THE XTAL VOLTAGE
C I IS THE COUNTER USED FOR FREQUENCY
C J IS THE COUNTER USED FOR POWER APPLIED
      PARAMETER(IMAX=514,IPMAX=10)
      DIMENSION FREQ(IMAX,IPMAX),POWIN(IMAX,IPMAX),POWREF(IMAX,IPMAX)
      DIMENSION VOLT(IMAX,IPMAX),IDRC(2),ITHERM(2),PWR(IPMAX)
      DIMENSION TEMPP(IMAX),TEMPF(IMAX)
      CHARACTER LABEL1*20,LABEL2*50,XLAB*15,YLAB*15,PTITLE*15
      CHARACTER FNAME*40
C GET THE DATA FILE NAME TO LOOK AT
      WRITE(*,*) 'ENTER THE DATA FILE NAME'
      READ(*,*) FNAME
      OPEN(UNIT=1,FILE=FNAME,STATUS='OLD')
      READ(1,*) LABEL1
      READ(1,*) IFLAG
      READ(1,*) LABEL1
      READ(1,*) IXTAL
      READ(1,*) LABEL1
      READ(1,*) IDRC(1)
      READ(1,*) LABEL1
      READ(1,*) IDRC(2)
      READ(1,*) LABEL1
      READ(1,*) ITHERM(1)
      READ(1,*) LABEL1
      READ(1,*) ITHERM(2)
      J=0
1     J=J+1
      READ(1,*,END=10) LABEL1
C PWR READS THE APPLIED POWER LEVEL
      READ(1,*) PWR(J)
      READ(1,*) LABEL2
      I=1
5     READ(1,*) FREQ(I,J),POWIN(I,J),POWREF(I,J),VOLT(I,J)
      IF(FREQ(I,J).EQ.-999.OR.POWIN(I,J).EQ.-999.OR.POWREF(I,J).EQ.
     +-999 .OR.VOLT(I,J).EQ.-999) GOTO 1
      I=I+1
      GOTO 5
10    CLOSE(UNIT=1)
      I=I-1
      J=J-1
      IF (I.GT.512) THEN
         WRITE(*,*) 'TO MANY DATA POINTS'
         STOP
      ENDIF
C REMOVE THERMISTOR RESPONSE FROM POWIN(I,J)
C AND POWREF(I,J) FOR J=1,2
```

```
      DO 15 K=1,J
         DO 12 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=POWIN(L,K)
12       CONTINUE
         ITH=ITHERM(1)
         CALL TCOR(I,TEMPF,TEMPP,ITH)
         DO 14 L=1,I
            POWIN(L,K)=TEMPP(L)
14       CONTINUE
15    CONTINUE
      DO 19 K=1,J
         DO 17 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=POWREF(L,K)
17       CONTINUE
         ITH=ITHERM(2)
         CALL TCOR(I,TEMPF,TEMPP,ITH)
         DO 18 L=1,I
            POWREF(L,K)=TEMPP(L)
18       CONTINUE
19    CONTINUE
C REMOVE THE DIRECTIONAL COUPLER RESPONSE
      DO 28 K=1,J
         DO 20 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=POWIN(L,K)
20       CONTINUE
         ID=IDRC(1)
         CALL DCCOR(I,TEMPF,TEMPP,ID)
         DO 24 L=1,I
            POWIN(L,K)=TEMPP(L)
24       CONTINUE
         DO 25 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=POWREF(L,K)
25       CONTINUE
         ID=IDRC(2)
         CALL DCCOR(I,TEMPF,TEMPP,ID)
         DO 26 L=1,I
            POWREF(L,K)=TEMPP(L)
26       CONTINUE
28    CONTINUE
C
29    WRITE(*,*) 'ENTER A 1 IF YOU WANT NET POWER COMPUTED'
      WRITE(*,*) 'ENTER A 0 IF YOU WANT INCIDENT POWER ONLY'
      READ(*,*) IOPT
      IF (IOPT.NE.1.AND.IOPT.NE.0) GOTO 29
C NOW THE POWER VALUES HAVE BEEN CORRECTED FOR THE FREQUENCY RESPONSE
C OF THE THERMISTOR MOUNTS AND THE DIRECTIONAL COUPLERS USED TO TAKE
C THE VARIOUS POWER VALUES.
C THE NET POWER WILL BE STORED IN POWIN, AND DATA FILES FOR BOTH THE
C APPLIED POWER AND THE INDUCED XTAL RESPONSE WILL BE GENERATED.
C SAVE NET POWER
      IF (IOPT.EQ.1) THEN
```

```
            DO 40 K=1,J
               DO 30 L=1,I
                  POWIN(L,K)=POWIN(L,K)-POWREF(L,K)
30             CONTINUE
40          CONTINUE
         ENDIF
C
C SET UP FOR GRAPHICS
C USE VAX GRAPHICS
C GRAPHICS FOR USE ON 4010,4014,4025,4105
         WRITE(*,*) 'ENTER THE DEVICE ID#'
         READ(*,*) IDEVIC
         IF (IDEVIC.EQ.4014.OR.IDEVIC.EQ.4105) THEN
            XMI=.01
            XMA=.9
            YMI=.01
            YMA=.9
         ELSEIF (IDEVIC.EQ.4010) THEN
            XMI=.01
            XMA=.6
            YMI=.01
            YMA=.6
         ELSEIF (IDEVIC.EQ.4025) THEN
            XMI=.05
            XMA=.85
            YMI=.05
            YMA=.85
            IDEVIC=4010
         ELSE
            WRITE(*,*) 'DEVICE ',IDEVIC,' NOT USED'
            STOP
         ENDIF
C PLOT POWER VS FREQ.
         CALL GSTART(IDEVIC)
         CALL NEWPAG
         XWMIN=FREQ(1,1)
         XWMAX=XWMIN
         YWMIN=POWIN(1,1)
         YWMAX=YWMIN
         DO 70 K=1,J
            DO 65 L=1,I
               TEMPF(L)=FREQ(L,K)
               TEMPP(L)=POWIN(L,K)
65          CONTINUE
            CALL MINMAX(TEMPF,I,X1,X2)
            CALL MINMAX(TEMPP,I,Y1,Y2)
            IF (XWMIN.GT.X1) XWMIN=X1
            IF (XWMAX.LT.X2) XWMAX=X2
            IF (YWMIN.GT.Y1) YWMIN=Y1
            IF (YWMAX.LT.Y2) YWMAX=Y2
70       CONTINUE
         CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
         CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
         CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
         XLAB='FREQ.(HZ)'
```

```
      YLAB='POWER(W)'
      IF (IOPT.EQ.1) PTITLE='NET POWER RESPONSE'
      IF (IOPT.EQ.0) PTITLE='INCIDENT POWER RESPONSE'
      CALL XLABEL(XLAB)
      CALL YLABEL(YLAB)
      CALL TITLE(PTITLE)
      DO 76 K=1,J
         DO 74 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=POWIN(L,K)
74       CONTINUE
         CALL CURVE(TEMPF,TEMPP,I,0,0)
76    CONTINUE
      CALL GRSTOP
78    READ(*,*) M
      IF (M.NE.1) GOTO 78
C PLOT VOLTAGE RESPONSE TO APPLIED POWER
      CALL GSTART(IDEVIC)
      CALL NEWPAG
      XWMIN=FREQ(1,1)
      XWMAX=XWMIN
      YWMIN=VOLT(1,1)
      YWMAX=YWMIN
      DO 85 K=1,J
         DO 80 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=VOLT(L,K)
80       CONTINUE
         CALL MINMAX(TEMPF,I,X1,X2)
         CALL MINMAX(TEMPP,I,Y1,Y2)
         IF (XWMIN.GT.X1) XWMIN=X1
         IF (XWMAX.LT.X2) XWMAX=X2
         IF (YWMIN.GT.Y1) YWMIN=Y1
         IF (YWMAX.LT.Y2) YWMAX=Y2
85    CONTINUE
      CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
      CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
      CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
      XLAB='FREQ.(HZ)'
      YLAB='VOLTS'
      PTITLE='XTAL FREQ. RESPONSE'
      CALL XLABEL(XLAB)
      CALL YLABEL(YLAB)
      CALL TITLE(PTITLE)
      DO 90 K=1,J
         DO 88 L=1,I
            TEMPF(L)=FREQ(L,K)
            TEMPP(L)=VOLT(L,K)
88       CONTINUE
         CALL CURVE(TEMPF,TEMPP,I,0,0)
90    CONTINUE
      CALL GRSTOP
C
C CREATE FILE FOR SAVING DATA
      OPEN(UNIT=1,FILE='XTALP.DAT',STATUS='NEW')
```

```
      OPEN(UNIT=2,FILE='XTALV.DAT',STATUS='NEW')
      WRITE(1,*) '''XTAL ID TAG'''
      WRITE(2,*) '''XTAL ID TAG'''
      WRITE(1,*) IXTAL
      WRITE(2,*) IXTAL
      WRITE(1,*) '''SWEEPER POWER OUTPUT'''
      WRITE(2,*) '''XTAL OUTPUT VOLTAGE'''
      WRITE(1,*) I,J
      WRITE(2,*) I,J
      WRITE(1,91) (PWR(K),K=1,J)
      WRITE(2,91) (PWR(K),K=1,J)
91    FORMAT(' ',''FREQ\POW'',2X,10(F8.2,5X))
      DO 95 L=1,I
         WRITE(2,92) FREQ(L,1),(VOLT(L,K),K=1,J)
         WRITE(1,92) FREQ(L,1),(POWIN(L,K),K=1,J)
95    CONTINUE
92    FORMAT(' ',11(1PE11.3,2X))
      CLOSE(UNIT=1)
      CLOSE(UNIT=2)
      STOP
      END
C
C SUBROUTINE TCOR
      SUBROUTINE TCOR(J,TEMPF,TEMPP,ITHERM)
      DIMENSION TEMPF(J),TEMPP(J)
      REAL*8 NU(50),CFAC(50),B(50),C(50),D(50),FRQ,CAL
      CHARACTER*20 FNAME
      IF (ITHERM.EQ.21593) FNAME='21593.MNT'
      IF (ITHERM.EQ.21594) FNAME='21594.MNT'
      IF (ITHERM.EQ.18832) FNAME='18832.MNT'
      IF (ITHERM.EQ.21627) FNAME='21627.MNT'
      IF (ITHERM.EQ.51009) FNAME='51009.MNT'
      IF (ITHERM.EQ.50918) FNAME='50918.MNT'
      IF (ITHERM.EQ.19131) FNAME='19131.MNT'
      OPEN(UNIT=2,FILE=FNAME,STATUS='OLD')
C GET THE NUMBER OF DATA POINTS
      READ(2,*) MAX
      DO 10 I=1,MAX
         READ(2,2) NU(I),CFAC(I)
2        FORMAT(2D10.2)
10    CONTINUE
      CLOSE(UNIT=2)
C CORRECT ARRAY TEMPP BY INTERPOLATING ARRAY CFAC
      CALL SPCOEF(MAX,NU,CFAC,B,C,D,IFLAG)
      IF (IFLAG.NE.0) THEN
         WRITE(*,*)'ERROR IN INTERPOLATION ',IFLAG
      ENDIF
      DO 20 I=1,J
         FRQ=DBLE(TEMPF(I))*(1.D-9)
         CALL SVALUE(MAX,NU,CFAC,B,C,D,FRQ,CAL,IFLAG)
         IF (IFLAG.NE.0) THEN
            WRITE(*,*)'ERROR IN EVALUATION ',IFLAG,TEMPF(I)
         ENDIF
         TEMPP(I)=TEMPP(I)/CAL
20    CONTINUE
```

```
      RETURN
      END
C
      SUBROUTINE DCCOR(J,TEMPF,TEMPP,ID)
      DIMENSION TEMPF(J),TEMPP(J)
      REAL*8 NU(514),PCKOFF(514),B(514),C(514),D(514),FRQ,VAL
      CHARACTER FNAME*20,LABEL*50
      IF (ID.EQ.1) FNAME='DC1.DAT'
      IF (ID.EQ.2) FNAME='DC2.DAT'
      IF (ID.EQ.3) FNAME='DC3.DAT'
      IF (ID.EQ.4) FNAME='DC4.DAT'
      OPEN(UNIT=2,FILE=FNAME,STATUS='OLD')
      READ(2,*) LABEL
      READ(2,*) I
      READ(2,*) LABEL
      READ(2,*) I
      READ(2,*) MAX
      READ(2,*) LABEL
      DO 5 I=1,MAX
         READ(2,*) A,Z
         NU(I)=DBLE(A)
         PCKOFF(I)=DBLE(Z)
5     CONTINUE
      CLOSE(UNIT=2)
C INTERPOLATE THE DATA
      CALL SPCOEF(MAX,NU,PCKOFF,B,C,D,IFLAG)
      IF (IFLAG.NE.0) THEN
         WRITE(*,*) 'ERROR IN INTERPOLATION',IFLAG
      ENDIF
C REMOVE DIR. COUPLERSE
      DO 10 I=1,J
         FRQ=DBLE(TEMPF(I))
         CALL SVALUE(MAX,NU,PCKOFF,B,C,D,FRQ,VAL,IFLAG)
         IF (IFLAG.NE.0) THEN
            WRITE(*,*)'ERROR IN PICKOFF EVALUATION',IFLAG,TEMPF(I)
         ENDIF
         TEMPP(I)=VAL*TEMPP(I)
10    CONTINUE
      RETURN
      END
```

PSWP.FOR

```
      PROGRAM PSWP
C THIS PROGRAM WILL READ THE OUTPUT FILES OF XTAL.FOR AND
C GENERATE A POWER SWEEP AT A REQUESTED FREQUENCY.
C IF THE FREQUENCY REQUESTED IS ONE ROW OF THE
C DATA FILE NO INTERPOLATION IS NECESSARY,
C IF THE FREQUENCY IS NOT THEN THE DATA IS LINEARLY INTERPOLATED
C BASED ON THE FREQUENCY DATA FOR THE ROW ABOVE AND BELOW.
      PARAMETER(IFR=514,IPWR=10)
      DIMENSION FREQ(IFR),POW(IFR,IPWR),VOLT(IFR,IPWR)
      DIMENSION VGPH(IPWR),PGPH(IPWR),VPLT(100),PPLT(100)
      REAL*8 T1(IFR),T2(IFR)
      REAL*8 BV(IFR),BP(IFR),CV(IFR),CP(IFR),DV(IFR),DP(IFR),VAL1,VAL2
      CHARACTER LABEL*40,FNAME*20,XLAB*10,YLAB*10,PTITLE*20
      WRITE(*,*) 'ENTER THE VOLTAGE DATA FILE NAME'
      READ(*,*) FNAME
      OPEN(UNIT=1,FILE=FNAME,STATUS='OLD')
      READ(1,*) LABEL
      READ(1,*) J
      READ(1,*) LABEL
C GET THE NUMBER OF ROWS AND COLUMNS IN THE DATA TABLE
      READ(1,*) IROW,ICOL
      READ(1,*) LABEL,(T1(I),I=1,ICOL)
      DO 5 I=1,IROW
         READ(1,*) FREQ(I),(VOLT(I,J),J=1,ICOL)
5     CONTINUE
10    CLOSE(UNIT=1)
      WRITE(*,*) 'ENTER THE POWER DATA FILE NAME'
      READ(*,*) FNAME
      OPEN(UNIT=1,FILE=FNAME,STATUS='OLD')
      READ(1,*) LABEL
      READ(1,*) J
      READ(1,*) LABEL
C GET THE NUMBER OF ROWS AND COLUMNS IN THE DATA TABLE
      READ(1,*) IROW,ICOL
      READ(1,*) LABEL,(T1(I),I=1,ICOL)
      DO 15 I=1,IROW
         READ(1,*) FREQ(I),(POW(I,J),J=1,ICOL)
15    CONTINUE
      CLOSE(UNIT=1)
C THE DATA TO BE ANALYZED IS NOW STORED IN THE ARRAYS POW AND VOLT
      WRITE(*,*) 'ENTER THE FREQUENCY(HZ) THAT YOU ARE INTERESTED IN'
      READ(*,*) F0
C FIND OUT IF WE NEED TO INTERPOLATE OR CAN WE USE THE DATA AT HAND
      IF (FREQ(1).GT.F0.OR.FREQ(IROW).LT.F0) THEN
         WRITE(*,*) 'THE FREQUENCY ',F0,' CAN NOT BE USED'
         STOP
      ENDIF
      DO 20 I=1,IROW
         IF (FREQ(I).LE.F0.AND.FREQ(I+1).GT.F0) ICH=I
20    CONTINUE
      WRITE(*,*) FREQ(ICH),F0,FREQ(ICH+1)
      IF (ABS(FREQ(ICH)-F0).GT.1.E-2) THEN
```

```
C TIME TO INTERPOLATE
C INTERPOLATE THE VOLTAGE
          DO 25 I=1,ICOL
             T1(1)=FREQ(ICH)
             T1(2)=FREQ(ICH+1)
             T2(1)=VOLT(ICH,I)
             T2(2)=VOLT(ICH+1,I)
             N=2
             CALL SPCOEF(N,T1,T2,BV,CV,DV,IFLAG)
             IF (IFLAG.LT.0) THEN
                WRITE(*,*) 'ERROR IN VOLTAGE INTERPOLATION'
                WRITE(*,*) IFLAG
                STOP
             ENDIF
             VAL1=F0
             CALL SVALUE(N,T1,T2,BV,CV,DV,VAL1,VAL2,IFLAG)
             IF (IFLAG.LT.0) THEN
                WRITE(*,*) 'ERROR IN EVALUATION'
                WRITE(*,*) IFLAG
                STOP
             ENDIF
             VGPH(I)=VAL2
25        CONTINUE
C INTERPOLATE THE POWER
          DO 30 I=1,ICOL
             T1(1)=FREQ(ICH)
             T1(2)=FREQ(ICH+1)
             T2(1)=POW(ICH,I)
             T2(2)=POW(ICH+1,I)
             N=2
             CALL SPCOEF(N,T1,T2,BP,CP,DP,IFLAG)
             IF (IFLAG.LT.0) THEN
                WRITE(*,*) 'ERROR IN POWER INTERPOLATION'
                WRITE(*,*) IFLAG
                STOP
             ENDIF
             VAL1=F0
             CALL SVALUE(N,T1,T2,BP,CP,DP,VAL1,VAL2,IFLAG)
             IF (IFLAG.LT.0) THEN
                WRITE(*,*) 'ERROR IN POWER EVALUATION'
                WRITE(*,*) IFLAG
                STOP
             ENDIF
             PGPH(I)=VAL2
30        CONTINUE
       ELSE
          DO 35 I=1,ICOL
             VGPH(I)=VOLT(ICH,I)
             PGPH(I)=POW(ICH,I)
35        CONTINUE
       ENDIF
C
C LET'S INTERPOLATE ON PGPH AND VGPH TO GET A SMOOTH CURVE
       DO 40 J=1,IPWR
          T1(J)=DBLE(VGPH(J))
```

```
         T2(J)=DBLE(PGPH(J))
40     CONTINUE
       CALL SPCOEF(ICOL,T1,T2,BP,CP,DP,IFLAG)
C
       STEP=(VGPH(ICOL)-VGPH(1))/99
       DO 45 J=1,100
          VPLT(J)=VGPH(1)+(J-1)*STEP
          VAL1=DBLE(VPLT(J))
          CALL SVALUE(ICOL,T1,T2,BP,CP,DP,VAL1,VAL2,IFLAG)
          PPLT(J)=SNGL(VAL2)
45     CONTINUE
C LET'S HAVE THE COMPUTER EVALUATE SOME XTAL VOLTAGES INTO POWERS
C BASED ON THE INTERPOLATED DATA POINTS
50     WRITE(*,*) 'ENTER THE XTAL VOLTAGE(VOLTS)'
       READ(*,*) ZVAL
       VAL1=DBLE(ZVAL)
       CALL SVALUE(ICOL,T1,T2,BP,CP,DP,VAL1,VAL2,IFLAG)
       WRITE(*,51) ZVAL,SNGL(VAL2),IFLAG
51     FORMAT(' ','FOR ',1PE10.3,' VOLTS, ',1PE10.3,' WATTS',I4)
52     WRITE(*,*) 'FOR MORE DATA ENTER A 1'
       WRITE(*,*) 'TO CONTINUE ENTER A 0'
       WRITE(*,*) 'TO STOP ENTER A 2'
       READ(*,*) ICNTR
       IF (ICNTR.EQ.1) GOTO 50
       IF (ICNTR.EQ.2) STOP
       IF (ICNTER.NE.0) GOTO 52
C WE ARE NOW READY TO PLOT PGPH VS VGPH
C USE VAX GRAPHICS
C GRAPHICS FOR USE ON 4010,4014,4025,4105
       WRITE(*,*) 'ENTER THE DEVICE ID#'
       READ(*,*) IDEVIC
       IF (IDEVIC.EQ.4014.OR.IDEVIC.EQ.4105) THEN
          XMI=.01
          XMA=.9
          YMI=.01
          YMA=.9
       ELSEIF (IDEVIC.EQ.4010) THEN
          XMI=.01
          XMA=.6
          YMI=.01
          YMA=.6
       ELSEIF (IDEVIC.EQ.4025) THEN
          XMI=.05
          XMA=.85
          YMI=.05
          YMA=.85
          IDEVIC=4010
       ELSE
          WRITE(*,*) 'DEVICE',IDEVIC,' NOT USED'
          STOP
       ENDIF
100    WRITE(*,*) 'ENTER A 1 FOR A LINEAR PLOT'
       WRITE(*,*) 'ENTER A 2 FOR A LOG-LOG PLOT'
       READ(*,*) IOPT
       IF (IOPT.NE.1 .AND. IOPT.NE.2) GOTO 100
```

```
CALL GSTART(IDEVIC)
CALL NEWPAG
CALL MINMAX(VGPH,ICOL,XWMIN,XWMAX)
CALL MINMAX(PGPH,ICOL,YWMIN,YWMAX)
IF (IOPT.EQ.1) THEN
   CALL MAP(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
   CALL XLIN(XWMIN,YWMIN,XWMIN,XWMAX,5)
   CALL YLIN(XWMIN,YWMIN,YWMIN,YWMAX,5)
ELSEIF (IOPT.EQ.2) THEN
   CALL MAPLL(XWMIN,XWMAX,YWMIN,YWMAX,XMI,XMA,YMI,YMA)
   CALL XLOG(XWMIN,YWMIN,XWMIN,XWMAX)
   CALL YLOG(XWMIN,YWMIN,YWMIN,YWMAX)
ENDIF
XLAB='VOLTS'
YLAB='POW(W)'
PTITLE='POWER SWEEP'
CALL XLABEL(XLAB)
CALL YLABEL(YLAB)
CALL TITLE(PTITLE)
CALL CURVE(VGPH,PGPH,ICOL,0,-1)
CALL CURVE(VPLT,PPLT,100,0,0)
CALL GRSTOP
STOP
END
```

END

DATE

FILMED

DTIC

JULY 88